ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

(Laboratorio di)
Amministrazione di sistemi

SNMP

Marco Prandini

Dipartimento di Informatica – Scienza e Ingegneria

Motivazioni – il problema

- Ogni apparato *managed* dispone di propri strumenti proprietari per configurazione e monitoraggio
  - per la configurazione persistente, sono inevitabili
    - ma le SDN stanno cambiando lo scenario, spostando il *control plane* fuori dai dispositivi
  - per il monitoraggio di base, sono un ostacolo all'automazione
    - tool inconsistenti
    - implementazioni firmware → non personalizzabili
  - protocolli di accesso generici → problemi
    - insicurezza del canale (es. TELNET)
    - accesso a funzionalità eccessive (shell interattiva)

Motivazioni – la soluzione

- SNMP
  - Simple Network Management Protocol
- Standardizza il modello dei dati
  - Proprietà di rete interessanti = oggetti definiti formalmente
  - esempio: la proprietà di un apparato di rete o di un calcolatore che definisce dov'è fisicamente collocato ha
    - identificativo univoco = 1.3.6.1.2.1.1.6
    - sintassi = stringa di (non più di 255) caratteri stampabili

      https://www.alvestrand.no/objectid/1.3.6.1.2.1.1.6.html
- Standardizza il modello di interazione
  - Protocollo applicativo per la comunicazione tra dispositivi ed entità che li gestisce
  - Veicolato su UDP

Il modello dei dati: OID

- Alla base: un modello generico per inquadrare qualsiasi oggetto concreto, proprietà di un oggetto, o concetto astratto
  - tramite un Object IDentifier (OID) come definito dallo standard X.660 dell'ITU
  - in una gerarchia globale che nasce da una radice anonima "." da cui discendono tre archi, due di competenza delle maggiori organizzazioni di standardizzazione + uno congiunto
    - 0: ITU-T
    - 1: ISO
    - 2: joint-iso-itu-t
  - I nodi hanno un identificativo numerico e uno simbolico
    - es. 1.3.6.1 == iso.identified-organization.dod.internet
- Esempi – navigazione online dell'albero degli OID
  - http://www.oid-info.com/cgi-bin/display?tree=
  - http://www.alvestrand.no/objectid/top.html

Il modello dei dati: MIB

- **Managed Information Base** è la collezione degli oggetti gestiti
  - da un apparato
  - da un sistema di monitoraggio
- Idealmente è la descrizione operativa dell'intero albero globale degli OID
  - in pratica è partizionato in subset (MIB modules)
- È in sostanza un catalogo che associa ad ogni oggetto
  - un OID
  - una sintassi (tipo di dato)
  - una codifica (descrizione della rappresentazione materiale per rendere possibile la comunicazione tra architetture diverse)
- Formalmente utilizza il linguaggio SMIv2, un sottoinsieme di ASN.1 definito dalle RFC 2578/2579

MIB

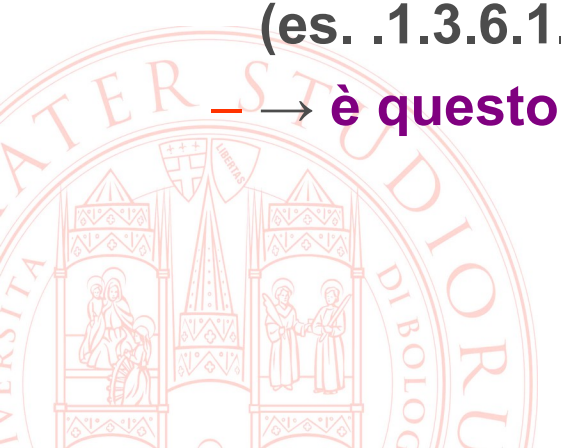
- Ogni entità interessata a descrivere le proprietà rilevanti in un certo contesto può definire un MIB.
- Es.
  - l'Internet Society definisce i protocolli di routing; nella RFC 4273 definisce identificatori, sintassi e codifiche per descrivere tutte le proprietà di BGP-4, come le rotte, gli eventi per aggiornarle, i peer che si scambiano le informazioni, ...
  - un produttore immette sul mercato un nuovo dispositivo, ad esempio una webcam, con supporto SNMP; pubblica il MIB che documenta identificatori, sintassi e codifiche per proprietà quali l'elenco delle risoluzioni supportate, la configurazione di rete, …
- Il modulo MIB relativo a un certo set di informazioni quindi deve essere noto a chi vuole interrogare un dispositivo che lo supporta, per sapere quali informazioni sono disponibili, e come vanno interpretate.

Il modello dei dati: sintassi

- Le sintassi supportate (SMIv1, SMIv2) sono:
  - simple data types
    - interi a 32 bit con segno
    - stringhe di byte (lunghezza massima 65.535)
    - OID
  - application-wide data types
    - *network addresses*; come IPv4, come generiche stringhe di byte
    - *counters*: interi a 32/64 bit positivi e crescenti, con rollover a zero
    - *gauges*: interi non negativi con limiti minimo e massimo
    - *time ticks*: centesimi di secondo trascorsi da un dato evento
    - *opaques*: stringhe arbitrarie senza controllo di sintassi
    - *integers*: ridefiniscono gli interi per avere precisione arbitraria
    - *unsigned integers*: come sopra ma senza segno
    - *bit strings*: stringhe di bit singolarmente identificati

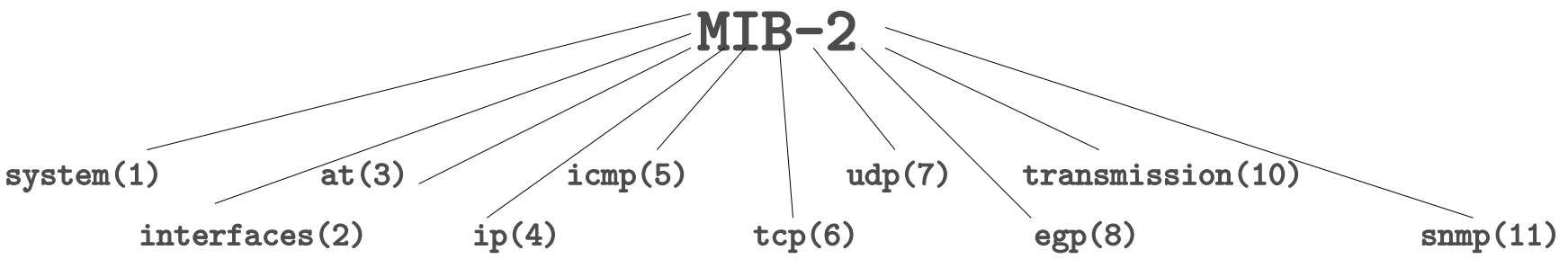
Il modello dei dati: scalari e tabelle

- Scalari e tabelle (array bidimensionali) sono le uniche strutture dati supportate
- Tre varianti sintattiche dell'OID
  - Un OID rappresenta in astratto il nodo dell'albero
  - Se una proprietà è **scalare**, es. il nome di un host (1.3.6.1.2.1.1.5) si aggiunge uno **zero** (1.3.6.1.2.1.1.5.0) per rappresentare l'istanza (a cui è associato il valore)
    → **è questo l'OID su cui materialmente operare letture e scritture**
  - Se una proprietà è una **tabella**, es. le interfacce di rete (1.3.6.1.2.1.2.2.1) si aggiunge `colonna.riga` (es. .1.3.6.1.2.1.2.2.1.3.2) per individuare la cella
  - → **è questo l'OID su cui materialmente operare letture e scritture**



MIB notevoli: il MIB-2

- Il modulo collocato sotto 1.3.6.1.2.1 è detto MIB-2
  - originariamente definito dalla RFC 1213
  - includeva tutti i dati essenziali per gli apparati di rete



  - da allora si sono staccati sotto-moduli per facilitarne l'estensione, ad esempio
    - TCP-MIB: RFC 4022 — Management Information Base for the Transmission Control Protocol (TCP)
    - UDP-MIB: RFC 4113 — Management Information Base for the User Datagram Protocol (UDP)
    - IP-MIB: RFC 4293 — Management Information Base for the Internet Protocol (IP)
    - IF-MIB: RFC 2863 — The Interfaces Group MIB

Esempi di managed object del MIB-2

- Gruppo system (1.3.6.1.2.1.1)
  - sysDescr(1)
  - sysContact(4)
  - sysName(5)
  - sysLocation(6)
- Gruppo IP (1.3.6.1.2.1.4)
  - 19 scalari
  - 4 tabelle
  - parametri generali del protocollo e tabelle con parametri specifici di ogni interfaccia, regola di routing, mappatura MAC
  - es. tabella ifEntry (1.3.6.1.2.1.2.2.1)

row ↓ col→	ifIndex (1)	ifDescr (2)	ifType (3)	ifMtu (4)	ifSpeed (5)	ifPhysAddress (6)									ifOutOctets (16)						ifSpecific (22)
(1)																					
(2)																					
…																					

MIB notevoli: private enterprise numbers (PEN)

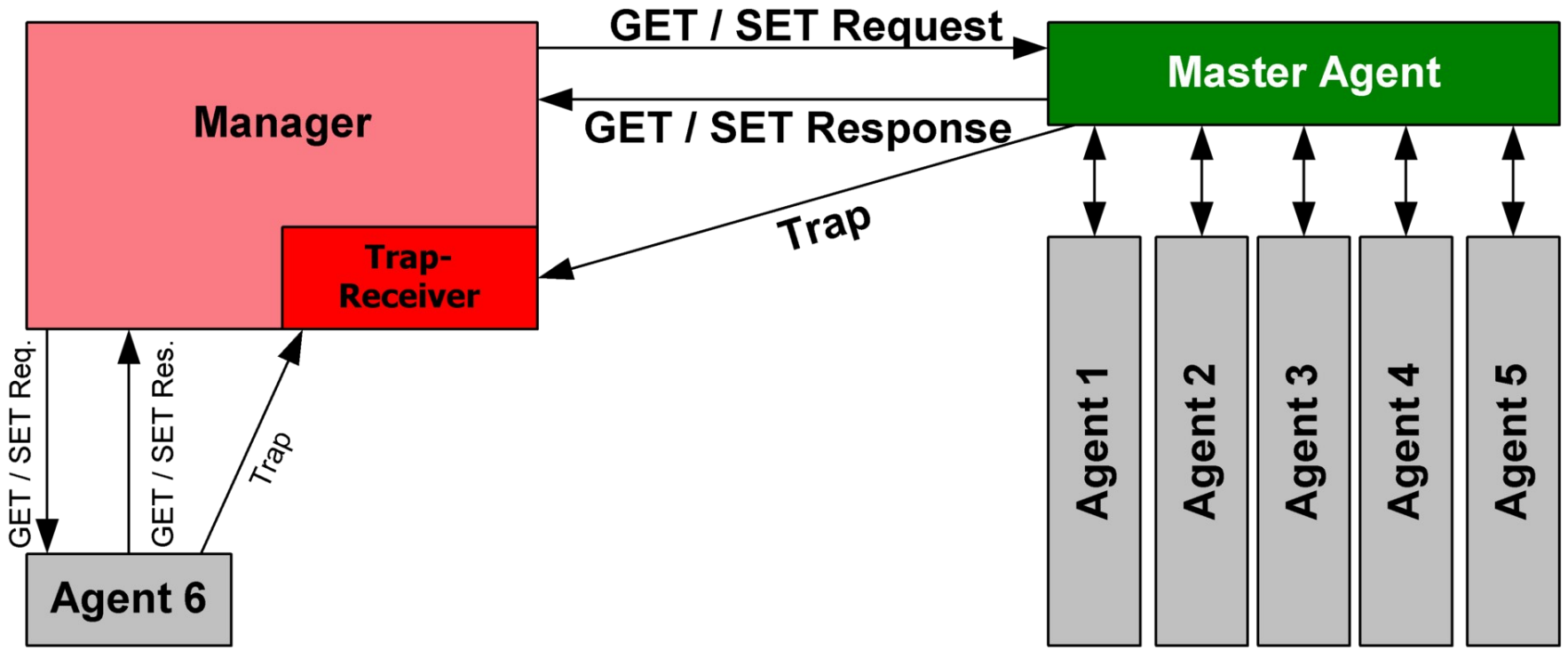
- Il sottoalbero 1.3.6.1.4.1 è dedicato a moduli specifici richiesti da enti privati (nel senso di non-ISO)
  - possono essere richiesti gratuitamente allo IANA http://pen.iana.org/pen/PenApplication.page
  - l'elenco è consultabile

    https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers
- Due PEN sono particolarmente significativi per il monitoraggio di sistemi operativi
  - estendono il MIB con oggetti generati dinamicamente
  - UCD-SNMP (1.3.6.1.4.1.2021)
    - accesso ai parametri base di un S.O. (stato dischi, memoria, processi, carico, log…)
  - NET-SNMP-EXTEND-MIB (1.3.6.1.4.1.8072)
    - output della direttiva *extend*
    - permette di trasformare l'output di qualsiasi script in un managed object

Il modello di interazione - definizioni

- I *managed object* sono le varie proprietà di un dispositivo, come finora descritte
- Il dispositivo prende il nome di *network element*
- Sul network element è in esecuzione un *agent*
  - software/firmware che accede a memoria e registri dei dispositivi fisici per rendere visibili i loro contenuti sotto forma di managed object
  - attraverso un protocollo di rete standard
- Il componente che accede agli agent è chiamato *manager*, tipicamente fa parte di un Network Management System (NMS)
- Il modello di interazione *manager-agent* è quindi simile ma non identico al modello *client-server*
  - manager ≈ client, agent ≈ server, ma con numerosità e risorse hardware invertiti
  - a volte l'agent prende l'iniziativa di contattare il manager

Il modello di interazione



Rene Bretz (updated by gh5046) - File:Snmp.PNG

SNMP communication principles diagram, changed one item from Deutsch to English. CC BY-SA 3.0

I protocolli

- SNMP è un protocollo a livello applicativo
  - trasportato su UDP
  - agent in ascolto su porta 161 (10161 variante su TLS/DTLS)
  - manager in ascolto su porta 162 (10162 variante su TLS/DTLS)
- Tre versioni "e mezzo"
  - v1, v2, v2c, v3
  - tutte accomunate dalla struttura del pacchetto (PDU)

version	community	PDU-type	request-id	error-status	error-index	variable bindings

I protocolli – tipi di PDU

GetRequest	richiede il valore associato a un managed object
SetRequest	richiede di settare il valore associato a un m.o.
GetnextRequest	richiede all'agent di scoprire qual è l'OID del m.o. successivo a quello specificato
GetbulkRequest	versione ottimizzata, che richiede di recuperare tutti gli oggetti successivi a quello specificato, fino a riempire un pacchetto UDP
Trap	notifica asincrona dall'agent al manager
InformRequest	notifica asincrona dall'agent al manager, con conferma di ricezione
Response	la risposta a uno dei precedenti comandi "Request" o "Trap"
Report	comunicazione inter-engine, principalmente per segnalare problemi con l'elaborazione di messaggi ricevuti

(rosso: solo v2 e v3; verde: solo v3)

I protocolli v1 e v2c - community

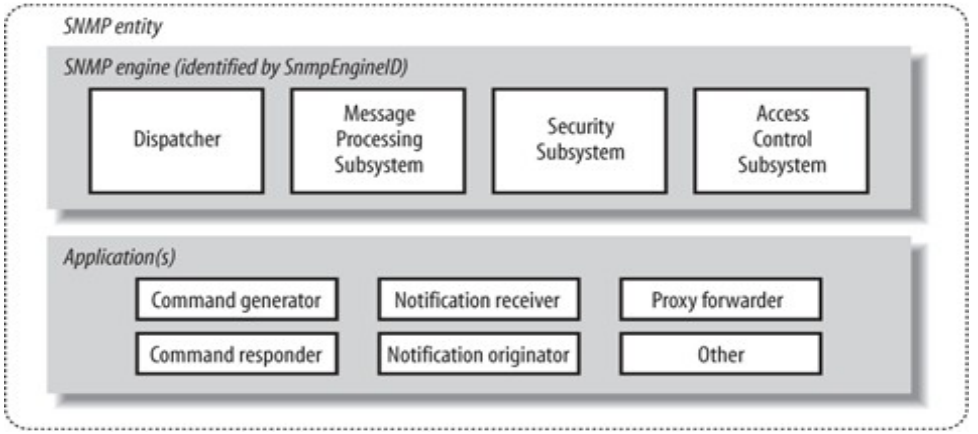
- Community = etichetta per stabilire il livello di fiducia tra manager e agent
- Tre livelli di autorizzazione
  - read-only
  - read-write
  - trap
- Ogni livello è identificato da una stringa
  - svolge insieme il ruolo di nome e password
  - se so che la read-write community di un agent si chiama *private*, questo è tutto ciò che mi serve per avere accesso in lettura e scrittura al suo MIB
- **Come metto in sicurezza una rete che condivide l'infrastruttura di gestione con quella operativa?**

I protocolli a confronto

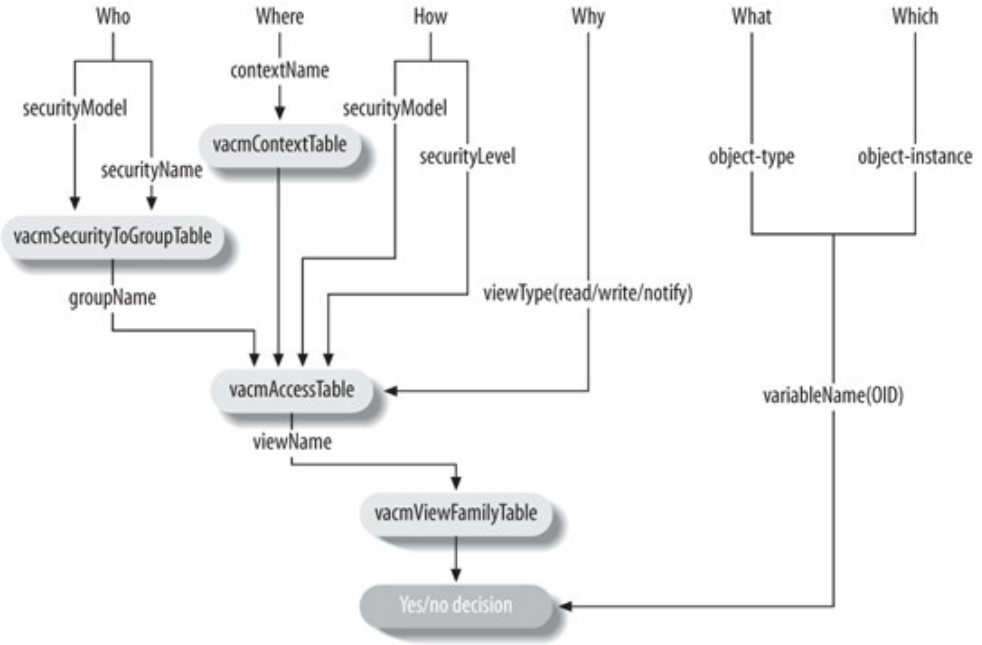
- SNMPv1
  - limitato a 32 bit
  - gestione errori minimale
- SNMPv2
  - nuovi data type, 64 bit
  - comandi GetBulk e Inform
  - party-based security system
  - macchinoso → scarsissima adozione
    - SNMPv2c (community = v2 senza party-based security)
    - SNMPv2u (user based, tentativo di migliorare la sicurezza rispetto a v1 senza complicare troppo, poi usato anche in v3)
- SNMPv3
  - vecchi comportamenti ridefiniti con termini totalmente diversi
  - niente più "manager" e "agent"
    - unificati sotto forma di SNMP entities
    - standardizzazione dell'architettura interna dei componenti
  - modello di sicurezza aggiornato

SNMPv3

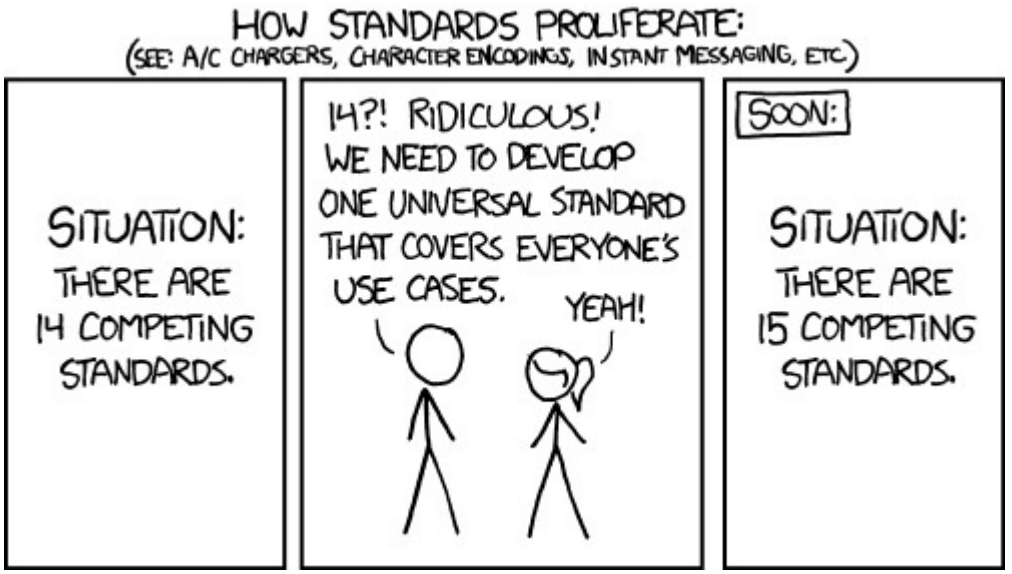
- Entity =
  - engine
    - rx/tx messaggi
    - estrazione dati
    - gestione sicurezza
  - applications
    - generazione delle richieste e delle notifiche
    - gestione delle risposte e delle notifiche



- Sicurezza:
  - User-based Security Model (USM): utenti autenticati con password protette da HMAC, canale cifrato con DES-3DES-AES
  - View-based Access Control Model (VACM):
    - utenti mappati su **gruppi**,
    - porzioni di MIB descritte come **vist**
    - matrice di controllo accessi (cosa può fare un **gruppo** su una **vista**)



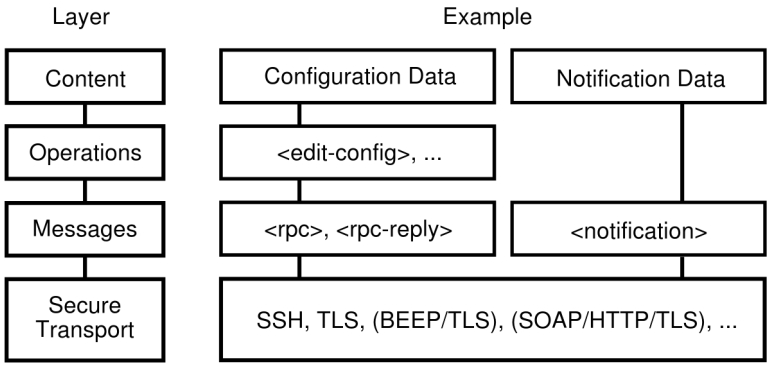
Alternative a SNMP

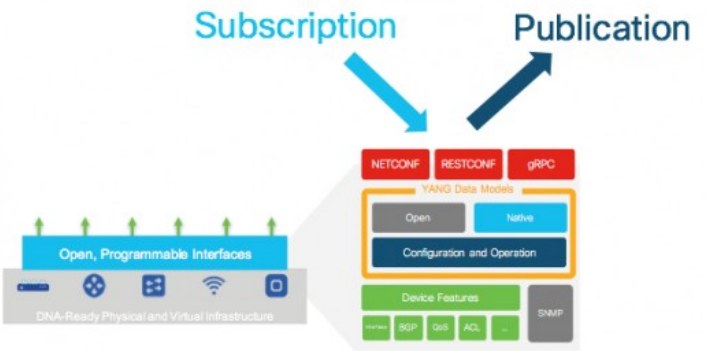


https://xkcd.com/927/

Alternative a SNMP

- Nuovi linguaggi per descrivere le informazioni
  - YANG (Yet Another Next Generation) – RFC 6020/6991/7950
  - strutture dati gerarchiche stile XML
  - indipendente dal protocollo di trasporto
- Nuovi protocolli per inviare configurazioni codificate (es. in YANG)
  - NETCONF – RFC 4741/6241 + estensioni
  - invio/modifica/rimozione di direttive via RPC
    - ed evoluzioni non standard come gRPC
  - transazionale multi-dispositivo
  - subset di funzionalità accessibili via HTTP (RESTCONF – RFC 8040)
- Cambio di paradigma: Model-Driven Telemetry
  - comunicazione publish-subscribe
  - via NETCONF/YANG si chiede al network element di inviare dati al/ai manager, periodicamente o quando si verificano determinate condizioni
- Microsoft fa a modo suo
  https://docs.microsoft.com/en-us/windows/win32/wmisdk/about-wmi





https://blogs.cisco.com/developer/its-time-to-move-away-from-snmp-and-cli-and-use-model-driven-telemetry

Network Management Systems

- SNMP (o le alternative) sono strumenti per scambiare informazioni
- un NMS li utilizza, ma in più rende disponibile all'amministratore una piattaforma per
  - organizzare l'inventario dei network element
    - template di dialogo per le diverse tipologie
    - raggruppamento per tipi, collocazioni, versioni, …
    - gestione credenziali
  - raccogliere i dati
    - polling vs. pub-sub
    - archiviazione, storicizzazione
  - visualizzare i dati
    - navigazione multidimensionale
    - reazioni a situazioni che necessitano attenzione

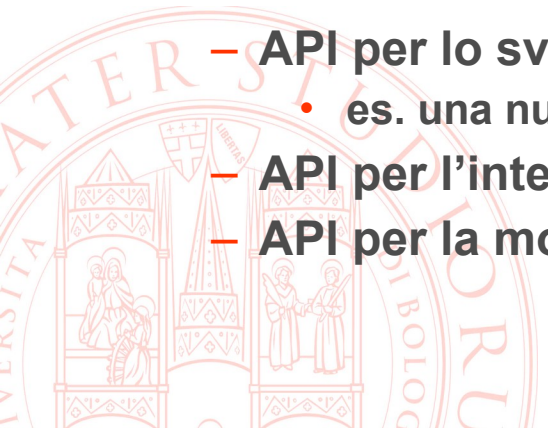
https://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems

Concetti base degli NMS

- item
  - il singolo dato da raccogliere
  - definito da tipo e metodo di accesso
  - tipicamente storicizzato
- trigger
  - una condizione valutata sugli item
  - negli NMS più complessi può consentire
    - test logici, aritmetici, pattern matching
    - valutazioni su serie storiche, ad es. soglia su ultime N osservazioni, media mobile, ...
    - combinazioni logiche di molteplici test di base
    - combinazioni di item
- action
  - la reazione innescata dall'attivazione di un trigger
- template
  - parametrizzate opportunamente, le definizioni degli item, dei trigger e delle action sono raccolte in "librerie"
  - un template ne individua e specializza un sottoinsieme adatto al monitoraggio di una determinata tipologia di network element

Funzioni avanzate degli NMS

- Capacità grafiche
  - sempre esistite, ormai date per scontate
- Capacità di auto-configurazione
  - rilevazione automatica di nuovi elementi e loro identificazione
  - inserimento nell'inventario e nella mappa di rete
  - applicazione del template corretto
  - attivazione della strategia di monitoraggio
- Componibilità
  - architetture HA
  - architetture gerarchiche
- Estendibilità
  - API per lo sviluppo di funzionalità aggiuntive
    - es. una nuova action, un nuovo algoritmo di aggregazione storica...
  - API per l'interrogazione dello stato
  - API per la modifica della configurazione



NMS costruiti su agent alternativi

- Approccio molto seguito:
  - realizzare un agent per varie piattaforme
  - inventare un protocollo per farlo parlare con un manager
- Esempi
  - https://www.zabbix.com/
  - https://prometheus.io/
  - https://github.com/netdata/netdata
  - https://www.monitorix.org/
  - https://github.com/statsd/statsd
  - https://www.influxdata.com/time-series-platform/telegraf/
  - https://collectd.org/
- Vantaggi: molti plug-in per raccogliere dati non formalizzati nei MIB esistenti
- Limiti: non poter installare un agent
  - piattaforma non supportata
  - piattaforma chiusa (es. embedded)

  → spesso il manager parla anche SNMP per integrare queste fonti